CLAIMS

What is claimed is:

1. A workflow system comprising:

a computer having at least one central processing unit (CPU);

a computer memory and/or storage, residing within said computer; and

a workflow software component, residing at least in part within said computer memory and/or storage, the workflow software component configured to execute a plurality of tasks to be performed automatically and configured to retry, for a predetermined number of times, to execute one of the plurality of tasks when said one of the plurality of tasks fails to be executed.

2. The system of claim 1, wherein the workflow software component is configured to process at least one short running service request among the plurality of tasks to be executed automatically, wherein the at least one short running service request is executed as a synchronous service.

3. The system of claim 1, wherein the workflow software component is configured to process at least one long running service request among the plurality of tasks to be executed automatically, wherein the at least one long running service request is executed as an asynchronous service.

4. The system of claim 1, wherein the predetermined number of times for said retry is equal to five.

5. The system of claim 1, wherein a first time interval between a first and a second retry is different from a second time interval between the second and a third retry.

6. The system of claim 1, wherein a time interval between a first and a second retry is shorter in duration than between any subsequent two consecutive retries.

7. The system of claim 1, wherein the workflow software component is configured to provide a standard software interface, thereby allowing an external software component to communicate therewith.

8. The system of claim 7, wherein the standard software interface complies with Component Object Model (COM).

9. The system of claim 1, wherein the workflow software component is further configured to commit a predetermined number of said plurality of tasks to be executed as a group.

10. A workflow system comprising:

a computer having at least one central processing unit (CPU);

a computer memory and/or storage, residing within said computer; and

a workflow software component, residing at least in part within said computer memory and/or storage, the workflow software component configured to execute a plurality of tasks to be performed automatically, wherein the workflow software component comprises:

a service provider configured to interface with at least one software object configured to carry out an instruction;

a task processor configured to execute the plurality of tasks by communicating with the at least one software object via the service provider; and

a process controller coupled to the task processor and configured to make a request to retry to execute one of the plurality of tasks when said one of the plurality of tasks fails to be executed by said task processor.

11. The system of claim 10, wherein the task processor is configured to attempt to lock another one of the plurality of tasks before said another one of the plurality of tasks is to be executed.

12. The system of claim 11, wherein the task processor is further configured to ensure that said another one of the plurality of tasks is not currently being executed when the task processor attempts lock the another one of the plurality of tasks.

13. The system of claim 10, further comprising:

a task initiator configured to make a request to the task processor to execute another one of the plurality of tasks, wherein the task processor executes said another one of the plurality of tasks in response to the request.

14. The system of claim 13, wherein the task initiator is further configured to retry to make the request to the task processor to execute the another one of the plurality of tasks when the task processor fails to executed the another one of plurality of tasks fails to be executed.

15. The system of claim 13, wherein the task controller is configured to make a request to the task initiator so that the another one of the plurality of tasks is executed by the task processor.

16. The system of claim 15, wherein the task processor is further configured to retry to make the request to the task initiator to execute the another one of the plurality of tasks when the another one of plurality of tasks fails to be executed.

17. The system of claim 10, wherein the instruction to be carried out by the service provider is to etch a lot of wafers.

18. The system of claim 10, wherein the workflow software component is configured to comply with Component Object Model (COM) objects.

19. The system of claim 10, wherein at least one of the plurality of tasks is a short running service having no return address in its Application Program Interface (API).

20. The system of claim 10, wherein at least one of the plurality of tasks is a long running service having a return address in its API, to thereby allow return information from the long running service is received by the return address.

21. The system of claim 20, wherein system resources are freed after the long running service has been called without waiting for the return information.

22. A workflow system comprising:

a computer having at least one central processing unit (CPU);

a computer memory and/or storage, residing within said computer; and

a workflow software means, residing at least in part within said computer memory and/or storage, the workflow software means configured to execute a plurality of tasks to be performed automatically, wherein the workflow software means comprises:

a service provider means for interfacing with at least one software object configured to carryout an instruction;

a task processor means for executing the plurality of tasks by communicating with the at least one software object via the service provider means; and

a process controller means for making a request to retry to execute one of the plurality of tasks when said one of the plurality of tasks fails to be executed by said task processor.

23. The system of claim 22, wherein the task processor means is configured to attempt to lock another one of the plurality of tasks before said another one of the plurality of tasks is to be executed.

24. The system of claim 23, wherein the task processor means is further configured to ensure that said another one of the plurality of tasks is not currently being executed when the task processor means attempts lock the another one of the plurality of tasks.

25. The system of claim 22, further comprising:

a task initiator means for making a request to the task processor means to execute another one of the plurality of tasks, wherein the task processor means executes said another one of the plurality of tasks in response to the request.

26. The system of claim 25, wherein the task initiator means is further configured to retry to make the request to the task processor means to execute the another one of the plurality of tasks when the task processor means fails to executed the another one of plurality of tasks fails to be executed.

27. The system of claim 25, wherein the task controller means for making a request to the task initiator means so that the another one of the plurality of tasks is executed by the task processor means.

28. The system of claim 27, wherein the task processor means is further configured to retry to make the request to the task initiator means to execute the another one of the plurality of tasks when the another one of plurality of tasks fails to be executed.

29. The system of claim 22, wherein the instruction to be carried out by the service provider means is to etch a lot of wafers.

30. The system of claim 22, wherein the workflow software means is configured to comply with Component Object Model (COM) objects.

31. The system of claim 22, wherein at least one of the plurality of tasks is a short running service having no return address in its Application Program Interface (API).

32. The system of claim 22, wherein at least one of the plurality of tasks is a long running service having a return address in its API, to thereby allow return information from the long running service to be received by the return address.

33. The system of claim 32, wherein system resources are freed after the long running service has been called without waiting for the return information.

34. A workflow method comprising the steps of:

(1) receiving a workflow script that includes a plurality of tasks configured to manufacture a product;

(2) automatically executing the plurality of tasks as defined in the workflow script; and

(3) retrying, for a predetermined number of times, to execute one of the plurality of tasks when the one of the plurality of tasks failed to be executed.

35. The method of claim 34, wherein the plurality of tasks of said step (1) comprises the step of including at least one short running service request, and wherein the method further comprises the step of:

synchronously executing the at least one short running service request.

36. The method of claim 34, wherein the plurality of tasks of said step (1) comprises the step of including at least one long running service request and wherein the method further comprises the step of:

asynchronously executing the at least one long running service request.

37. The method of claim 34, wherein said step (3) comprises the step of:

retrying at least five times when the one of the plurality of tasks continue to fail to be executed.

38. The method of claim 34, wherein said step (3) comprises the step of:

configuring a first time interval between a first and a second retry to be different from a second time interval between the second and a third retry.

39. The method of claim 34, wherein the retrying step includes the step of :

configuring a time interval between a first and a second retry to be shorter than between any subsequent two consecutive retries.

40. The method of claim 34, further comprising the step of:

committing a predetermine number of the plurality of tasks to be executed as a group.

41. A computer readable medium including instructions being executed by a computer, the instructions instructing the computer to create and use a computer-implemented workflow, the instructions comprising implementation of the steps of:

(1) receiving a workflow script that includes a plurality of tasks configured to manufacture a product;

(2) automatically executing the plurality of tasks as defined in the workflow script; and

(3) retrying, for a predetermined number of times, to execute one of the plurality of tasks when the one of the plurality of tasks failed to be executed.

42. The medium of claim 41, wherein the plurality of tasks of said step (1) comprises the step of including at least one short running service and wherein the method further comprises the step of:

synchronously executing the at least one short running service request.

43. The medium of claim 41, wherein the plurality of tasks of said step (1) comprises the step of including at least one long running service request and wherein the method further comprises the step of:

asynchronously executing the at least one long running service request.

44. The medium of claim 41, wherein said step (3) includes the step of:

retrying at least five times when the one of the plurality of tasks continue to fail to be executed.

45. The medium of claim 41, wherein said step (3) includes the step of:

configuring a first time interval between a first and a second retry to be different from a second time interval between the second and a third retry.

46. The medium of claim 41, wherein said step (3) includes the step of :

configuring a time interval between a first and a second retry to be shorter than between any subsequent two consecutive retries

47. The medium of claim 41, further comprising the step of:

committing a predetermine number of the plurality of tasks to be executed as a group.

48. A workflow system comprising:

a computer having at least one central processing unit (CPU);

a computer memory and/or storage, residing within said computer; and

a workflow software component, residing at least in part within said computer memory and/or storage, the workflow software component configured to execute a plurality of tasks to be

performed automatically and configured to retry, for a predetermined number of times, to execute one of the plurality of tasks when said one of the plurality of tasks fails to be executed,

wherein the workflow software component is configured to process at least one short running service request and at least one long running service among the plurality of tasks to be executed automatically, and

wherein the at least one short running service request is executed as a synchronous service and the at least one long running service request is executed as an asynchronous service.

49. A workflow system comprising:

a computer having at least one central processing unit (CPU);

a computer memory and/or storage, residing within said computer; and

a workflow software component, residing at least in part within said computer memory and/or storage, the workflow software component configured to execute a plurality of tasks to be performed automatically and configured to retry, for a predetermined number of times, to execute one of the plurality of tasks when said one of the plurality of tasks fails to be executed,

wherein a first time interval between a first and a second retry is different from a second time interval between the second and a third retry.

50. A workflow method comprising the steps of:

(1) receiving a workflow script that includes a plurality of tasks configured to manufacture a product, wherein

(i) synchronously executing at least one short running service request, wherein the plurality of tasks comprises the at least one short running service request; and

(ii) asynchronously executing at least one long running service request, wherein the plurality of tasks comprises the at least one long running service request;

(2) automatically executing the plurality of tasks as defined in the workflow script;

(3) retrying, for a predetermined number of times, to execute one of the plurality of tasks when the one of the plurality of tasks failed to be executed; and

(4) configuring a time interval between a first and a second retry to be shorter than between any subsequent two consecutive retries.